What is claimed is:

1. In a packet-switched communication system including a plurality of data sources and a plurality of destinations, interconnected by a plurality of communication switches,

an integrated shaping-scheduling mechanism responsive to shaped packet flows from a first set of data sources and to unshaped packet flows from a second set of data sources, for providing a single, multiplexed output flow of data packets from both the first and the second set of sources, the mechanism comprising:

a receiver for organizing incoming packets from said plurality of data sources into a plurality of predetermined flow queues, a first partition of flow queues corresponding to shaped packet flows and a second partition of flow queues corresponding to unshaped packet flows, each shaped packet flow being associated with a separate traffic profile, said traffic profile being defined as means for expressing upper limits on the amount of data packets that the packet communication system is allowed to transmit for an associated shaped packet flow;

a plurality of traffic regulators, each traffic regulator being used for determining whether a corresponding flow is classified as either “virtually compliant” or “virtually incompliant”, where all unshaped flows are always virtually compliant, and shaped flows are virtually compliant as long as their traffic profiles are not close to be violated, and virtually incompliant when their traffic profiles are close to be violated;

a selector for determining the order of transmission of data packets associated with packet flows, the activity of the selector being driven by the values of a plurality of state variables associated with the packet flows;

a flow-dequeue element for removing a just-transmitted packet from its queue and modifying one or more of said plurality of state variables associated with that queue; and

a flow-enqueue element for updating one or more of said plurality of state variables when a new packet reaches the head of one of a plurality of packet queues, each of said packet queues being associated with a respective packet flow.

2. A packet-switched communication system as defined in claim 1, wherein the traffic profile of a shaped flow f_i is expressed by a set of dual-leaky-bucket traffic-

regulator parameters, said dual-leaky-bucket traffic-regulator parameters being called a sustainable-bit-rate (SBR) bucket size $B_{i,s}$, a sustainable-bit-rate token rate $\rho_{i,s}$, a peak-bit-rate (PBR) bucket size $B_{i,p}$, and a peak-bit-rate token rate $\rho_{i,p}$.

3. A packet-switched communication system as defined in claim 2, wherein the status of virtual compliance of a packet flow f_i is determined by comparison of said dual-leaky-bucket traffic-regulator parameters with a plurality of associated state variables, said state variables being called token levels $X_{i,s}$ and $X_{i,p}$, and time of latest update τ_i.

4. A packet-switched communication system as defined in claim 3, wherein the selector determines the virtually-compliant versus virtually-incompliant status of a packet flow f_i using the following test:

$$X_{i,p} \quad \leq \quad B_{i,p} - \left(\frac{l_i^k}{\rho_{i,p}} - \frac{l_i^k}{r} \right)$$

$$X_{i,s} \quad \leq \quad B_{i,s} - \left(\frac{l_i^k}{\rho_{i,s}} - \frac{l_i^k}{r} \right)$$

wherein l_i^k is the length of the data packet currently located at the head of the packet queue associated with the packet flow being tested, said data packet being referred to as the head-of-the-queue packet of said packet queue, and r is the total transmission capacity of the server, the packet flow being classified as "virtually compliant" as long as both inequalities are satisfied, and as "virtually incompliant" otherwise.

5. A packet-switched communication system as defined in claim 1, wherein the selector determines the next packet to be transmitted by first searching for eligible packet flows and transmitting a packet from a queue associated with one of such eligible packet flows if any is found, and if no eligible packet flow is found, the selector determines the next packet to be transmitted by searching for a virtually-compliant packet flow and transmitting a packet from a queue associated with one of such virtually-compliant packet

flows if any is found, and if no virtually-compliant packet flow is found the selector determines no packet transmission.

6. A packet-switched communication system as defined in claim 5, wherein a timestamp is associated with each of a plurality of packet flows, each timestamp being used in the determination of the eligibility and virtual-compliance status of the associated packet flow and in the determination of the order of transmission of packets.

7. A packet-switched communication system as defined in claim 6, wherein the eligibility of a packet flow is determined by checking the associated timestamp in accordance with a Smallest Eligible Finishing time First (SEFF) packet-selection policy.

8. A packet-switched communication system as defined in claim 7, wherein the flow-dequeue element modifies the state variables associated with the packet queue of the packet flow of a just-transmitted packet with the assessment of a new eligibility status of the packet flow and the consequent setting of an "eligibility flag" associated with the packet flow.

9. A packet-switched communication system as defined in claim 8, wherein a timestamp is recalibrated by the flow-dequeue element if a data packet has just been transmitted from the packet queue of the associated packet flow and the current value of said timestamp is greater than the current "real" time plus the service interval of the just transmitted packet, the recalibrated timestamp being equal to the sum of the "real" time plus the service interval of the just transmitted packet, said service interval being equal to the ratio between the length of the just transmitted packet and the guaranteed service rate of the packet flow of the just transmitted packet.

10. A packet-switched communication system as defined in claim 8, wherein the flow-enqueue element uses the result of the virtual-compliance test applied to the packet flow being processed and the current value of the eligibility flag of said packet flow to compute a value for the timestamp associated with said packet flow.

11. A packet-switched communication system as defined in claim 10, wherein if the value in the eligibility flag is "FALSE" and the associated packet flow is virtually incompliant, the generated timestamp is the maximum of either the previous timestamp or the current "real" time plus the service interval of the head-of-the-queue packet of said packet flow, with said service interval being added to said maximum.

12. A packet-switched communication system as defined in claim 10, wherein if the head-of-queue packet of a packet flow is a newly-arrived packet, the generated timestamp is the maximum of either the previous timestamp of the packet flow or the current "real" time, with the value of the service interval of the head-of-the-queue packet being added to said maximum.

13. A packet-switched communication system as defined in claim 10, wherein if the packet flow is virtually-compliant and its head-of-the-queue packet did not just arrive to the queue, the generated timestamp is equal to the previous timestamp of the packet flow plus the service interval of the head-of-the-queue packet.

14. A method of scheduling service over a plurality of queues holding flows of packet data traffic, the method integrating shaping of packet flows and scheduling of both shaped and unshaped packet flows and defined by the steps of:

responsive to receiving a plurality of data packets from a plurality of data sources, identifying for each receiving data packet a respective queue for storage of said data packet;

storing each received data packet in its identified queue;

determining, for each shaped packet flow, a set of traffic-regulator parameters associated with a traffic profile of said packet flow;

for each packet flow, performing a "virtual-compliance" test every time a new packet reaches the head of the corresponding flow queue, wherein each unshaped flow is predetermined as being always classified as virtually compliant, and each shaped flow is determined to be classified as virtually compliant as long as the size of its head-of-queue

packet does not create the possibility of violating the predetermined traffic-regulator parameters;

selecting a packet for transmission from among eligible packet flows and virtually-compliant packet flows, therefore always excluding from the selection data packets of packet flows that are both non-eligible and virtually-incompliant;

performing a flow-dequeue process subsequent to data packet transmission for recalibrating a plurality of state variables associated with the packet queue of the packet flow of the just transmitted data packet; and

performing a flow-enqueue process upon every arrival of a new packet to the head of a corresponding flow queue.

15. The method as defined in claim 14, wherein the traffic regulators are defined by dual-leaky-bucket parameters for a sustainable-bit-rate (SBR) leaky bucket and a peak-bit-rate (PBR) leaky bucket, and include for each leaky bucket a first state variable X that defines the number of tokens in the bucket, and a second state variable τ that expresses the time when the number of tokens was last updated.

16. The method according to claim 15, wherein the test for virtual compliance is based on the verification of the following inequalities:

$$X_{i,p} \leq B_{i,p} - \left(\frac{l_i^k}{\rho_{i,p}} - \frac{l_i^k}{r} \right)$$

$$X_{i,s} \leq B_{i,s} - \left(\frac{l_i^k}{\rho_{i,s}} - \frac{l_i^k}{r} \right)$$

wherein $B_{i,s}$ and $B_{i,p}$ are the "bucket sizes" for the two leaky buckets of the packet flow being tested, l_i^k is the length of the head-of-the-queue packet of said packet flow, $\rho_{i,s}$ and $\rho_{i,p}$ are the token rates of the two leaky buckets, and r is the total transmission capacity of the server, the packet flow being defined as "virtually compliant" as long as both inequalities are satisfied.

17. The method according to claim 14, wherein in determining the order of packets to be transmitted, the following steps are used:

searching for an eligible packet flow, and selecting an eligible packet flow with the shortest time before expiration of the transmission deadline for its head-of-the-queue packet, said transmission deadline being expressed by a timestamp associated with said packet flow; if no such eligible packet flow exists, then

searching for a virtually-compliant packet flow, and selecting a virtually-compliant packet flow with the shortest time before expiration of the transmission deadline for its head-of-the-queue packet; if no such virtually-compliant packet flow exists, then

not selecting any packet for transmission at that time.

18. The method according to claim 14, wherein the eligibility of a packet flow is determined in accordance with a Smallest Eligible Finishing time First (SEFF) packet-selection policy.

19. The method according to claim 18, wherein in performing the flow-dequeue process for a packet flow f_i, the following steps are executed to determine the value of an eligibility flag EF_i and possibly recalibrate the timestamp F_i^k of the packet flow:

if $\left(F_i^k > t_{i,d}^k + l_i^k / r_i\right)$ /* Timestamp is not eligible */

$EF_i \leftarrow FALSE$

$F_i^k \leftarrow t_{i,d}^k + l_i^k / r_i$ /* Recalibration of timestamp */

else

$EF_i \leftarrow TRUE$ /* Timestamp is eligible */

where $t_{i,d}^k$ is the current "real" time.

20. The method according to claim 19, wherein in performing the flow-enqueue process for a packet flow f_i, the following steps are executed:

if $\left(EF_i = FALSE\right)$ **and** $\left(i \in I\right)$

$$F_i^k \leftarrow \max\left(F_i^{k-1}, t_{i,h}^k + l_i^k / r_i\right) + l_i^k / r_i$$

else if (flow 22-i is newly backlogged)

$$F_i^k \leftarrow \max\left(F_i^{k-1}, t_{i,h}^k\right) + l_i^k / r_i$$

else

$$F_i^k \leftarrow F_i^{k-1} + l_i^k / r_i$$

where I is defined as the set of virtually-incompliant packet flows.